Amendments to the Claims

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of claims:

1. to 37. (cancelled)

38. (new) A method for retrieving data related to versions of files organized in a configuration, said method comprising:

-arranging a proxy device between a server device and one or more clients;

-providing a proxy cache at said proxy device for storing bulk data and meta data related to said versions of files, said bulk data comprising said versions themselves, said meta data comprising information about the organization and properties of said files into a versioned system;

-registering said proxy cache with a list of listeners maintained by said server device in order to update said proxy cache when changes are made at said server device to said data related to versions of files, said changes being controlled by only said server device;

-enabling said proxy cache to receive an update from said server device according to changes made to said data related to versions of files;

-enabling said proxy cache to be updated using said update to thereby synchronize said proxy cache with said server device;

-enabling said proxy device to provide said update to said one or more clients for synchronizing one or more client-based caches;

-receiving a request from one of said clients for one or more desired file versions or a set of meta data;

-if said one or more desired file versions or set of meta data is in said proxy cache, returning said one or more desired filed versions or set of meta data to said one of said clients; and

-if said one or more desired file versions or set of meta data is not in said proxy cache, said proxy device forwarding said request to said server device, obtaining from said server device said one or more desired file versions or set of meta data, and returning to said one of said clients said one or more desired file versions or set of meta data.

39. (new) The method in claim 38 further comprising said proxy device presenting a credential to said server device to authenticate said proxy device before said proxy cache is enabled to receive said update from said server device.

40. (new) The method in claim 38 wherein upon receiving said request from said one of said clients for said desired file version or set of meta data, said proxy cache examines said set of meta data to determine if said one of said clients has permission to access said desired file version or set of meta data.

41. (new) The method in claim 38 wherein said configuration of said data related to versions of files comprises a tree structure, said tree structure comprising a plurality of successive versions and one or more alternate revisions.

42. (new) The method in claim 41 wherein one or more changes between any two versions of files are stored as one or more deltas, and said desired file version is reconstructed by said server device using said one or more deltas.

43. (new) The method in claim 38 wherein each of said one or more clients has a sandbox for locally storing said data related to versions of files.

44. (new) The method in claim 38 further comprising said proxy device forwarding a lock request to said server device on behalf of said one of said clients, said lock request for said desired file version and, if granted by said server device, permitting only said one of said clients to modify said desired file version.

45. (new) A computer readable medium comprising computer executable instructions that when executed by a processor at a proxy device perform the method according to claim 38.

46. (new) A method for retrieving data related to versions of files organized in a configuration, said method comprising:
    -arranging a server device in communication with one or more proxy devices arranged to be between said server device and one or more clients, each of said one or more proxy device

comprising a proxy cache;

 - providing a repository at said server device for storing multiple versions of the same file and meta data comprising information about the organization and properties of said files into a versioned system, said server device controlling all changes to data in said repository;

 -providing a server cache at said server device for storing bulk data and meta data related to said versions of files, said bulk data comprising said versions themselves, said meta data comprising said information about the organization and properties of said files into a versioned system;

 -registering at said server device respective proxy caches for said one or more proxy devices with a list of listeners maintained by said server device in order to update said proxy caches when changes are made at said server device to said data related to versions of files, said changes being controlled by only said server device;

 -enabling said server device to transmit updates to said respective proxy caches according to changes made to said data related to versions of files according to what is currently stored by a respective one of said proxy caches to synchronize said proxy caches;

 -receiving a request from one of said proxy devices on behalf of one of said clients for one or more desired file versions or a set of meta data when said request cannot be fulfilled by said one of said proxy devices;

 -if said one or more desired file versions or set of meta data is in said server cache, returning said one or more desired filed versions or set of meta data to said one of said proxy devices; and

 -if said one or more desired file versions or set of meta data is not in said server cache, said server device obtaining said one or more desired file versions or set of meta data using said repository, and returning said one or more desired file versions or set of meta data using to said one of said proxy devices to enable said proxy device to fulfill said request made by said one of said clients.

47. (new) The method in claim 46 further comprising said proxy device presenting a credential to said server device to authenticate said proxy device before said proxy cache is enabled to receive said update from said server device.

48. (new) The method in claim 46 wherein upon said proxy device receiving said request from said one of said clients for said desired file version or set of meta data, said proxy cache

examines said set of meta data to determine if said one of said clients has permission to access said desired file version or set of meta data.

49. (new) The method in claim 46 wherein said configuration of said data related to versions of files comprises a tree structure, said tree structure comprising a plurality of successive versions and one or more alternate revisions.

50. (new) The method in claim 49 wherein one or more changes between any two versions of files are stored as one or more deltas and said server device reconstructing said desired file version using said one or more deltas when obtaining said desired file version from said repository.

51. (new) The method in claim 46 wherein each of said one or more clients has a sandbox for locally storing said data related to versions of files.

52. (new) The method in claim 46 further comprising said server device receiving a lock request from said one of said proxy device on behalf of said one of said clients, said lock request for said desired file version and, if granted by said server device, permitting only said one of said clients to modify said desired file version.

53. (new) A computer readable medium comprising computer executable instructions that when executed by a processor at a server device perform the method according to claim 46.

54. (new) A system for retrieving data related to versions of files organized in a configuration, said system comprising:

-a proxy device positioned between a server device and one or more clients, said proxy device comprising a processor and a proxy cache for storing bulk data and meta data related to said versions of files, said bulk data comprising said versions themselves, said meta data comprising information about the organization and properties of said files into a versioned system, said proxy cache registered with a list of listeners maintained by said server device in order to update said proxy cache when changes are made at said server device to said data related to versions of files, said changes being controlled by only said server device, said proxy cache configured to receive an update from said server device according to changes made to

said data related to versions of files, said proxy cache configured to be updated using said update to thereby synchronize said proxy cache with said server device, said proxy device configured to provide said update to said one or more clients for synchronizing one or more client-based caches, said proxy device configured to receive a request from one of said clients for one or more desired file versions or a set of meta data, and said processor configured such that:

-if said one or more desired file versions or set of meta data is in said proxy cache, said proxy device returns said one or more desired file versions or set of meta data to said one of said clients; and

-if said one or more desired file versions or set of meta data is not in said proxy cache, said proxy device forwards said request to said server device, obtains from said server device said one or more desired file versions or set of meta data, and returns to said one of said clients said one or more desired file versions or set of meta data.

55. (new) The system in claim 54 wherein said proxy device is configured to present a credential to said server device to authenticate said proxy device before said proxy cache is enabled to receive said update from said server device.

56. (new) The system in claim 54 wherein upon receiving said request from said one of said clients for said desired file version or set of meta data, said proxy cache is configured to examine said set of meta data to determine if said one of said clients has permission to access said desired file version or set of meta data.

57. (new) The system in claim 54 wherein said configuration of said data related to versions of files comprises a tree structure, said tree structure comprising a plurality of successive versions and one or more alternate revisions.

58. (new) The system in claim 57 wherein one or more changes between any two versions of files are stored as one or more deltas, and said desired file version is reconstructed by said server device using said one or more deltas.

59. (new) The system in claim 54 wherein each of said one or more clients has a sandbox for locally storing said data related to versions of files.

60. (new) The system in claim 54 wherein said proxy device is able to forward a lock request to said server device on behalf of said one of said clients, said lock request for said desired file version and, if granted by said server device, permitting only said one of said clients to modify said desired file version.

61. (new) A system for retrieving data related to versions of files organized in a configuration, said system comprising:

-a server device in communication with one or more proxy devices arranged in between said server device and one or more clients, each of said one or more proxy devices comprising a proxy cache, said server device comprising a processor and a repository for storing multiple versions of the same file and meta data comprising information about the organization and properties of said files into a versioned system, said server device controlling all changes to data in said repository, said server device also comprising a server cache for storing bulk data and meta data related to said versions of files, said bulk data comprising said versions themselves, said meta data comprising information about the organization and properties of said files into a versioned system; said server device registering respective proxy caches for said one or more proxy devices with a list of listeners maintained by said server device in order to update said proxy caches when changes are made at said server device to said data related to versions of files, said changes being controlled by only said server device; said server device configured to transmit updates to said respective proxy caches according to changes made to said data related to versions of files according to what is currently stored by a respective one of said proxy caches to synchronize said proxy caches; said server device configured for receiving a request from one of said proxy devices on behalf of one of said clients for one or more desired file versions or a set of meta data when said request cannot be fulfilled by said one of said proxy devices; and said processor configured such that:

-if said one or more desired file versions or set of meta data is in said server cache, returning said one or more desired filed versions or set of meta data to said one of said proxy devices; and

-if said one or more desired file versions or set of meta data is not in said server cache, said server device obtaining said one or more desired file versions or set of meta data using said repository, and returning said one or more desired file versions or set of meta data using to said one of said proxy devices to enable said proxy device to fulfill said request made by said

one of said clients.

62. (new) The system in claim 61 wherein said server device is configured to receive a credential from said proxy device to authenticate said proxy device before said proxy cache is enabled to receive said update from said server device.

63. (new) The method in claim 61 wherein upon said proxy device receiving said request from said one of said clients for said desired file version or set of meta data, said proxy cache is configured to examine said set of meta data to determine if said one of said clients has permission to access said desired file version or set of meta data.

64. (new) The system in claim 61 wherein said configuration of said data related to versions of files comprises a tree structure, said tree structure comprising a plurality of successive versions and one or more alternate revisions.

65. (new) The system in claim 64 wherein one or more changes between any two versions of files are stored as one or more deltas and said server device reconstructing said desired file version using said one or more deltas when obtaining said desired file version from said repository.

66. (new) The system in claim 61 wherein each of said one or more clients has a sandbox for locally storing said data related to versions of files.

67. (new) The system in claim 61 wherein said server device is able to receive a lock request from said one of said proxy device on behalf of said one of said clients, said lock request for said desired file version and, if granted by said server device, permitting only said one of said clients to modify said desired file version.